N P S
R E S E A R C H

Volume 8, Number 3 *October 1998*

AN ARCHITECTURE FOR DYNAMIC PLANNING AND EXECUTION USING LOOSELY COUPLED COMPONENTS

Professor Gordon H. Bradley
Visiting Assistant Professor Arnold H. Buss
Military Instructor LTC Charles H. Shaw, III, USA
Department of Operations Research

Background

The Loosely Coupled Components Project is a faculty/student research effort that is designing, developing, and validating a software architecture for advanced military planning and execution. The architecture supports the rapid construction of systems for planning and execution that operate seamlessly over a global network of heterogeneous computing devices and software systems. The architecture coordinates a collection of components that operate on a computer network (for example, the Internet, NIPRNet, or the SIPRNet) to access unit or location data, maps, overlays, algorithms and other information. The components perform tasks such as: displaying maps, satellite images, and overlays; accessing, entering, and modifying data; constructing and displaying models of military operations; and accessing and executing algorithms to analyze operations. The components are designed and constructed independently of each other and they can be combined rapidly and inexpensively to build a wide variety of tools for military planning and execution. The design allows systems to be easily extended by adding additional components.

The fundamental research question is how to effectively use emerging commercial off-the-shelf (COTS) information technology to build advanced military systems for planning and execution with the capabilities envisioned in Joint Vision 2010, the Air Force's New Worlds Vista, the Navy's Quantum Leap, Army XXI Advanced Concept Technology Demonstrations (ACTDs), and other studies of warfare in the next century. The potential for the military is clear. However, like the technology advances of the past (steam powered ships, telegraph, radio, tanks, planes), it is not clear precisely how the technology can be incorporated into military planning and operations. It is even less clear how strategy and tactics need to change to best utilize the new capabilities. The United States has a definite advantage because of our economic resources and access to new technology. However, COTS technology, by its very nature, is also available to our potential adversaries. And military history is replete with examples of sudden shifts in relative military power because one military was able to more quickly understand the most effective way to use new technologies (for example, the tank and blitzkrieg, the plane and carrier based aviation, or radar and air defense).

--continued on page 2

IN THIS ISSUE

NPS RESEARCH
is published by the Assistant Dean of Research in accordance with NAVSO P-35. Views and opinions expressed are not necessarily those of the Department of the Navy. Comments/inquiries can be addressed via email to research@nps.navy.mil.

FEATURED PROJECT

ARCHITECTURE, *continued from page 1*

The post Cold War era is characterized by a wider spectrum of possible military missions, from Major Theater War (MTW) to Small Scale Contingencies (SSC) to Peacekeeping Operations (PKO), and by greater uncertainty about when and where military response will be required. This has generated requirements for flexible planning tools that support rapid response to situations whose details cannot be anticipated. These requirements are extensive. The planning resources (people, data, computers) will be distributed over a disparate, global network that has a range of computers (from supercomputers to cellular phones) and different software (operating systems, databases). The decision cycles will be much shorter, meaning the systems planning and execution need to work much faster with less time to make and review plans. Furthermore, there is a requirement to provide automatic monitoring of the planning and execution processes. The problems faced by planners will be less predictable than in the past, so the systems must be more flexible to address situations the designers cannot anticipate. The systems must have an open architecture that allows additional capabilities to be added without disruption. Legacy systems for planning and execution are too static, monolithic, and inflexible to meet these requirements. Current efforts to integrate legacy planning tools are an improvement, but, even when these efforts are brought to fruition, the results will not be sufficiently interoperable, platform independent, or extensible to meet the challenges of military decision making. As demanding as the individual requirements are, advanced systems for planning and execution must incorporate all these capabilities in an integrated system.

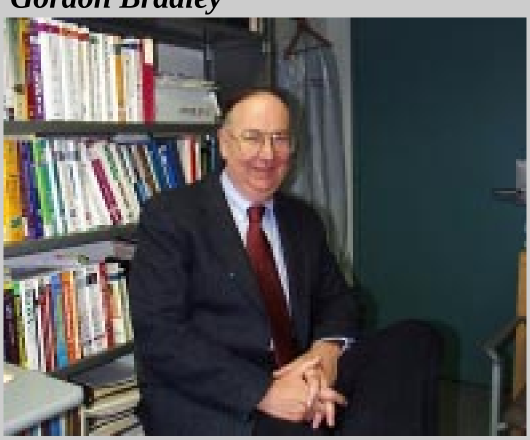
-- *continued on page 3*

About the INVESTIGATORS

Gordon H. Bradley is a Professor of Operations Research. His research interests are in mathematical programming and software development. He has worked on integer programming and network optimization. Currently he is working on the design and development of a software architecture for dynamic map-based military systems for planning and execution using new platform-independent technologies. He teaches the Operations Research Department's Java course and advanced seminars in applying information technology to operations research problems.

Dr. Bradley received his Ph.D. from Northwestern University in 1967 followed by postdoctoral study at Stanford University. His first academic appointment was at Yale University where he was Assistant and then Associate Professor of Operations Research with a joint appointment in Computer Science. He joined the Operations Research Department at the Naval Postgraduate School in 1973. From 1977 to 1987 he was Professor of Computer Science including five years as Chair. Since 1987 his appointment has been in the Department of Operations Research.

Gordon Bradley



Arnold Buss

Arnold H. Buss has taught and performed research in simulation analysis for eleven years. For the past four years he has been a Visiting Professor in Operations Research at the Naval Postgraduate School, where he has taught simulation and computer programming and supervised a number of student theses emphasizing simulation modeling.

Professor Buss has conducted research in component-based simulation modeling and simulation output analysis, as well as in manufacturing, project management and capacity planning. Current work includes simulation modeling in Java,

-- *continued on page 3*

ARCHITECTURE, *continued from page 2*

Design of the Loosely Coupled Components Architecture

The basic architecture group is composed of the authors plus **MAJ Arent Arntzen**, Norway AF (MS in OR, September 1998) and **MAJ Leroy Jackson**, USA, US Army TRADOC Analysis Center-Monterey (Ph.D. candidate in OR). Each member of the group is working on different components; the group meets weekly to develop and refine the architecture and to review the design of the components. The group studies the emerging research on software components. The design proceeds by adopting where possible and developing where necessary the principles that underlie the design. The architecture specifies how individual components will interact (through mediators) and how components will be composed to produce larger components. At every level of composition, the components have the same interface to other components. The programming language choice, Java, offers many advantages. Java is a good object-oriented language that supports composition through its "interface" construct and it contains powerful reflection and remote method execution capabilities that are used heavily in our work. The group studied Java's component technology, called Java Beans, very carefully, but ultimately decided it was not adequate for the dynamic composition that is necessary for our systems. We thus were lead to develop our own component design, which we have refined and validated by constructing several demonstration systems.

This component design currently consists of three major parts: the König package for creating graphs and interfaces for running algorithms on the graphs; an Application Programming Interface (API) for visually representing

-- *continued on page 31*

INVESTIGATORS,
continued from page 2

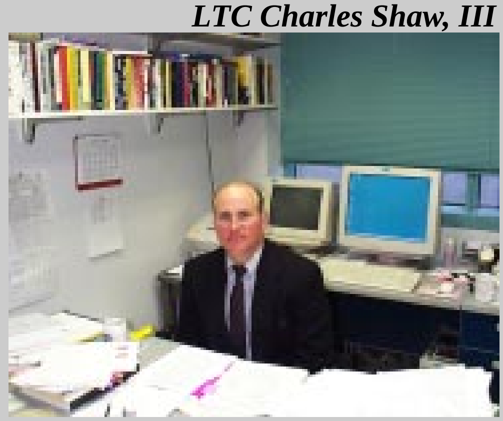
distributed simulation on the Internet, and component standards and frameworks for simulation modeling. He is a member of INFORMS, POMS, and IIE. He is an Associate Editor for *IIE Transactions* and is on the Editorial Review Board for *Production and Operations Management*. He has published papers in journals such as *Operations Research, IIE Transactions, Decision Sciences, Mathematical Problems in Engineering*, and the *Journal of Wind Engineering and Industrial Aerodynamics*.

Professor Buss received his B.A. from Rutgers University, M.S. degrees from the University of Arizona and Cornell University, and a Ph.D. from Cornell University. He is a member of Phi Beta Kappa, and was the recipient of the McMullen Graduate Fellowship and the Sage Graduate Fellowship while at Cornell University.

Charles H. Shaw, III is a Lieutenant Colonel, Quartermaster, in the United States Army currently serving as a Military Instructor in the Department of Operations Research and Fellow with the Institute for Joint Warfare Analysis (IJWA). His research and teaching interests include Joint and Combined Operations, Combat Modeling and Simulation, Joint and Combined Operational Logistics, Special Operations, Logistics Distribution Systems, and Military Operations Other Than War (MOOTW). He has served in this capacity since 1995.

LTC Shaw received a B.S. in Engineering from the U.S. Military Academy at West Point, NY in 1979 at which time he was commissioned as a Lieutenant in the Armor/Cavalry Branch. He received a M.S. in Operations Research from the Naval Postgraduate School at Monterey, CA in 1989. Prior to his current assignment, LTC Shaw served as the Battalion Executive Officer/Deputy Commander, 193rd Combat Support Battalion (Airborne); Commander, 193rd Brigade Material Management Center; Chief, Logistics Plans and Operations Division, J4; and Chief, Readiness and Force Integration Branch, J4 in the U.S. Army South and Joint Task Force - Panama located at Fort Clayton, Republic of Panama. Other previous assignments include Senior Operations Research Analyst, Defense Logistics Agency at Cameron Station, VA before, during, and after Operations Desert Shield/Storm and multiple assignments as an Armor/Cavalry Officer and Logistician at the tactical level in the United States, Germany, and Middle East.

LTC Charles Shaw, III



ARCHITECTURE, *continued from page 3*

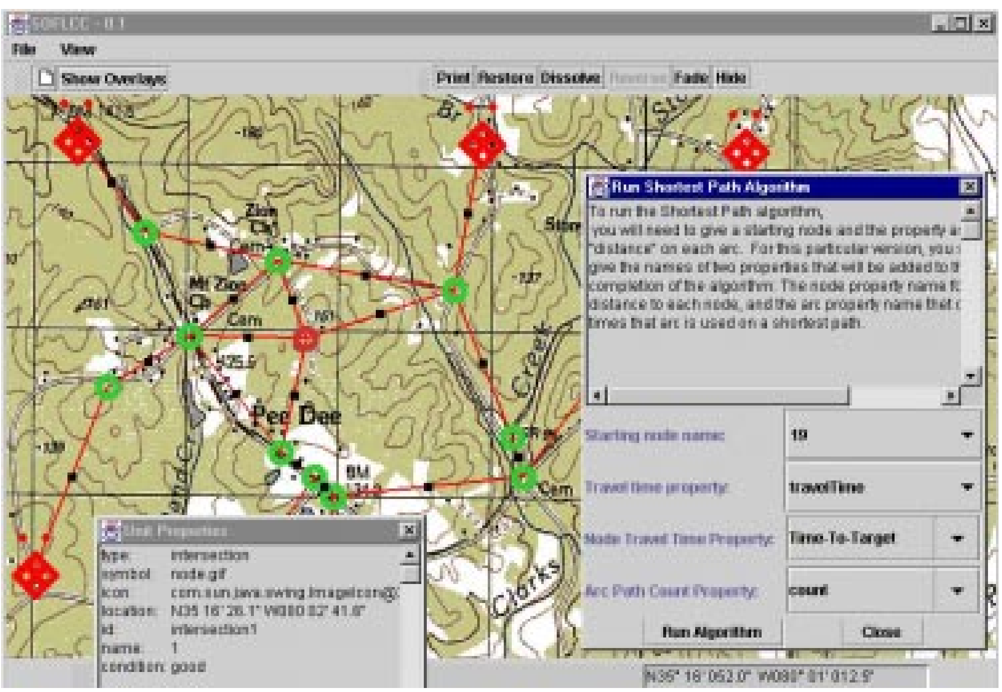
graphs and overlaying them on maps; and facilities for dynamically loading and running algorithms on the König graph objects. MAJ Jackson developed the König component as part of his PhD research. König contains a toolkit of methods to quickly construct graph and network algorithms. It also contains methods to load, display, and manipulate graphs and networks. A fourth part is currently not represented in the system, but perhaps best embodies the distillation of the lessons we have learned so far. This part consists of a standard for components and is implemented in the Modkit package. MAJ Arntzen is the primary author of this standard; it was developed with much intense discussion and experimentation by the group. MAJ Arntzen's thesis used Modkit to create rapid prototypes of Discrete-Event Simulation models of Integrated Air Defense Systems (IADS) that utilized both active and passive sensors with the goal of evaluating the contribution of the passive sensors. He was recently awarded the MORS/Tisdale prize for the best military Operations Research master's thesis (see Student Research, page 18).

The Modkit component standard represents a substantial advance over current component technologies such as Java Beans. It supports dynamic, recursive composition and effectively enables the users of conforming components to rapidly assemble extremely complex models from simple building blocks. Modkit's component standard will form the conceptual basis for the next-generation system, which has an anticipated release six months hence.

Applications of the Research

The research approach has been to augment our conceptual work by developing several Advanced Concept Technology Demonstrations (ACTDs) at NPS based on our architecture in order to demonstrate and validate our approach. This empirical approach has allowed us to rapidly test or fast prototype our ideas. It has also allowed us to use these working prototype systems to demonstrate the design and to show the capabilities that emerging information technology can bring to military planning. Our first demonstration project is the thesis work of **LT Sean Moriarty**, USN, (MS in OR September 1997) who was in the Operational Logistics curriculum. The system is a prototype advanced planning system for interdiction and restoration of logistics lines of communication. The system can load a map locally or download one from anywhere on the Internet. The user constructs a distribution network, i.e., transportation, power, communications, etc., by using the mouse to specify a network of nodes and arcs over the map. Properties can be specified for the nodes and arcs (for example, cost, capacity). The system can invoke an algorithm to determine an optimal allocation of forces to arcs in order to disconnect the nodes that are the sources of goods from designation nodes. In his thesis, LT Moriarty described three possible scenarios for his system: interdiction of supply routes (as was done in Viet Nam and more recently in the interdiction of drugs), the protection of a distribution network from attack, and an Operations Other the War (OOTW) mission to optimally apply resources to restore water after a natural

-- *continued on page 32*



Special Operations Forces Planning and Execution Prototype—computing the time for enemy units to engage Navy Seals who are attacking a radar site.

ARCHITECTURE, *continued from page 31*

disaster. Because the system is written in Java, it executes without modification on a wide variety of computing devices and environments. The project demonstrated that using easily available technology, a military planning system that downloaded resources from a network could be quickly constructed.

Immediately after the completion of the first system, the basic architecture group assessed the strengths and weaknesses of the system and after three months of research and design began work on the second generation, which resulted in a system to do dynamic planning and execution for Special Operations. This system includes an improved mapping component, the graph component developed by MAJ Jackson, and several algorithms developed by MAJ Arnsten using his Modkit component standard. The system was constructed to address a Special Operations scenario developed by **CPT Allan Bilyeu**, USA (MS in OR, June 1998) that is described in his thesis. The Special Operations Forces (SOF) scenario involves the attack on a radar site in Bosnia by a Navy Seal team operating in the Adriatic Sea. Air Force Special Forces operating out of Italy transport the Seals to the site. Army Special Forces working with coalition forces and NATO troops support the Seals on the ground. This map-based system supports downloading from a network various resources such as maps, satellite images, overlays of forces, roads, etc., and algorithms for use in developing plans in real time to support the mission. The loosely coupled components architecture allows these resources to be loaded as needed and to be incorporated dynamically into the executing system. The platform independence that Java provides is very critical to SOF because their planning involves units from different services, other government agencies, coalition forces, and international agencies spread around the world. It is unlikely they will be using identical computers or operating systems. Our system for planning and execution provides a common operating picture and similar functionality to units that are geographically dispersed and are using different hardware and software.

Planning does not cease with the onset of a mission's execution, but must continue as the mission unfolds. A planning system must therefore be capable of immediately reacting to any unanticipated events. As the mission proceeds, operators rather than planners become increasingly involved in the planning/replanning process. Thus, systems for planning and execution must provide seamless support of a common operating picture for both operators and planners. In the operations environment, the available computing devices are smaller (cellular phones, palmtops, etc.) and have fewer resources than the computing platforms available to the planners. A common operating picture must be maintained even as the active participants in the system change. Our architecture and system software was designed from the beginning to support capabilities across the widest possible range of computing devices from super-computers to the emerging small, lightweight battlespace devices. To that end, the kernel system was designed with the smallest possible computational footprint (currently less than 1 MB). By taking advantage of the relative large bandwidth from satellite to ground unit and the capability of our planning system to load resources as needed and when needed, small computing devices have sufficient capabilities to present an operational picture and to support local computation to do continuous replanning.

The dynamic capabilities of the system are best shown by considering how algorithms are located, loaded, and executed. The user selects a graph embedded in an overlay and then selects an algorithm (for example, shortest path or max flow) to execute on it. Any algorithm may be selected to run on any graph at the user's discretion. Since Java has dynamic loading, algorithms need not be loaded until they are needed. A class containing an algorithm can be loaded from the local machine or over a network. Once a class is selected, the system loads it using its name (a string). Utilizing a Java capability called reflection, the names of the algorithms in the class are discovered and presented to the planner. After the planner selects an algorithm, the system uses reflection on the algorithm to determine the parameters needed to execute it. For example, in a shortest path algorithm, the parameters are a source node and an arc property for the length. The system presents the planner with a short description of the algorithm and drop-down menus to select which of the nodes is the source and which arc property should be used for the length (there could be several choices, for example, length via road or length via air). The system then executes the algorithm and places the solution in the graph in the form of new or modified properties. For example, in a shortest path algorithm, the user may decide that each node should have a property called "Time-To-Target" that would represent the length of the shortest path from the node to the target. When the

-- continued on page 33

ARCHITECTURE, *continued from page 32*

algorithm is executed, that property is added to each node. Since the algorithm is loaded dynamically, it need not be known at compile time or even at the onset of the planning session. In an extreme example, an analyst at a remote location could be writing an algorithm while the planning is going on and, as soon as the algorithm is completed, it can be located, loaded into the planning system, and executed on a graph model. Although the general capabilities for dynamic behavior are built into Java, it should be noted that some specific capabilities are achieved only by using the loosely coupled components architecture that we have developed over the past year.

We have taken the prototype SOF system for planning and execution from design to the development of a working system in less than seven months; thus we have been able to give live demonstrations of the capabilities of a system based on our loosely coupled components architecture. This has allowed us to effectively present the work to high level personnel who may not have the technical background to understand the details of our design. Furthermore, these rapid cycles have allowed us to apply lessons learned from each application of the system to the design of the next generation.

Another recent application constructed with our architecture and components is the Coordinated Inland Area Search and Rescue (SAR) System (COINSS), developed as part of the thesis of **LT Timothy Castle**, USCG (MS in OR, September 1998). COINSS is a system to support the execution of searches for missing people in the continental United States. It advances the state-of-the-art by constructing a component that accounts for the movement of the target after the search begins, by integrating the calculations with a map and graphical user interface, and by implementing the software so that it can be executed without modification on a wide variety of computers. COINSS is a set of computational algorithms that compute a probability distribution of the location of the target over the search region. As the search proceeds, the distribution is updated to account for the movement of the target. The movement is calculated by applying a Markov process to a movement vector of velocities that depends on the type of target (for example, a child, as opposed to an adult hiker). The system can include elevation data; the system currently uses Digital Terrain Elevation Data Level 1 (DTED1) from the National Imagery and Mapping Agency (NIMA). Computer Science Research Assistant Professor **Wolfgang Baer** assisted us in accessing this information. The elevation data is used to compute the slope of the ground, which is then used to modify the movement of the target. The search coordinator assigns areas to various search teams. After each unsuccessful search of an area, the probability distribution is updated using Bayes theorem. COINSS was developed as a separate component, which was then integrated with the map, overlay, graph and algorithm components described above in the SOF application. The map component displays a map and provides latitude and longitude information; the overlay and graph components allow the search region and the search assignments to be input by mouse clicks, and the algorithms component makes the connection between the visual display and COINSS. The probability distribution is displayed over the map in shades of red to indicate the probabilities. The search coordinator (and any one monitoring the search) can see at a glance the changing search probabilities as the target moves and as search teams report back negative results. The integration of COINSS with the other components was done quickly because components in the loosely coupled design know very little about each other. The map layer displays images without knowing if they are maps, satellite pictures, graphs or probability distributions. To the system kernel, COINSS is just an application that has algorithms to invoke and an overlay to display. A significant benefit of the design is the interchangeability of components. Although the COINSS application was designed to find lost people in the United States, it can be invoked from the SOF application to find a downed pilot in Bosnia or track a fleeing war criminal.

Another application that uses the components is the thesis research of **LT Scott Schwartz**, USN (MS in OR, September 1998) that constructs a system to solve graph partition problems. The focus of his thesis is a problem from the Defense Information Systems Agency (DISA) to determine the optimal sequence to upgrade a communications network that has more than 200 nodes. The problem is modeled as the partition of a graph into a given number of subgraphs. Because there are no effective algorithms to construct an optimal solution, the system allows an analyst to construct a good (but not necessarily optimal) solution by dynamically specifying a sequence of heuristic algorithms to be applied to the problem. Because there are too many nodes and arcs to be displayed effectively, this

-- continued on page 34

ARCHITECTURE, *continued from page 33*

application is not map-based. However, the system employs the graph component to construct the heuristics and the algorithm component to dynamically load and execute the algorithms.

Presentations and Research Sponsors

In the past year the authors have given eleven presentations of the results in classified settings at the National Security Agency (NSA) and Headquarters, United States Special Operations Command (USSOCOM), and in unclassified presentations at professional meetings in the United States and abroad. We have also given a number of demonstrations to NPS visitors and in Washington.

The demonstration systems have been of particular interest to our research sponsors because they have provided early proof that advanced systems for planning and execution can be built using Java and that these systems can access resources over a network of heterogeneous computer assets. We have shown that we can dynamically locate, access and execute algorithms, and thus build "thin clients" that load programs if and when needed. This is particularly important for the emerging small, lightweight battlespace computing devices. The loosely coupled design shows how new components can be added quickly to existing systems. The research has been of interest to the general operations research community because it demonstrates that traditional analysis tools that have been used off-line can be integrated into real time systems.

The basic research on the architecture is supported by a 6.1 grant from the Air Force Office of Scientific Research (AFOSR) to Bradley and Buss. The AFOSR program is funded under an initiative based on the Air Force's New Worlds Vista study which "identified those technologies that will guarantee the air and space superiority of the United States in the 21st century." The research on graph and network algorithms is supported by a 6.1 grant from the Office of Naval Research (ONR) to Bradley, and to Professors **Jerry Brown** and **Kevin Wood**. The participation of MAJ Jackson is supported by the U.S. Army TRADOC Analysis Center (TRAC) - Monterey. NPS' Institute for Joint Warfare Analysis (IJWA) has provided support to LTC Shaw to develop the Special Operations planning tools and scenarios. The IJWA funds have been matched by the USSOCOM. USSOCOM has supported the development of realistic contemporary and future joint military planning scenarios by allowing students to spend

Graduation September 1998: Professor Gordon Bradley, LT Timothy Castle, LT Scott Schwartz, Major Leroy Jackson, Major Arent Arntzen, LTC Charles Shaw, Visiting Assistant Professor Arnold Buss

six weeks at their headquarters at MacDill AFB before they begin their thesis research.

The authors are advising USSOCOM on the development of their Mission Planning, Analysis, Rehearsal and Execution (MPARE) System. MPARE is the management and oversight process that will guide the integration and use of constructive simulations and computer based operational tools to enhance combat capability of special operations forces. GEN Schoomaker, CinC USSOCOM, has designated MPARE the command's "flagship" system. We are members of the Requirements Integration Program Team (RIPT) with responsibilities to contribute to the development of the Concept of Operations (CONOP) and Operational Requirements Documents (ORD) which are currently being written. We are also responsible to provide information on technological trends in C4I and possible "leap ahead" technologies.

Ongoing Research

We have begun the development of the third generation of demonstration systems based on our loosely coupled architecture. Our rapid design/development cycles match those of the commercial technologies that we use. In high technology, COTS development times are measured in so-called "internet time," with systems being developed and deployed in months rather than the years that is typical of many DoD software projects. While our architecture and user interface remain stable, it is easy to extend our systems

-- continued on page 43

ARCHITECTURE, *continued from page 34*

as new military threats emerge and new technologies are introduced. The continuing research will develop and demonstrate systems for planning and execution that present all the participants a common operating picture in a seamless system that operates from planning through execution and that executes on computers in the planning activity and in the battlespace.

The live demonstrations of our systems for planning and execution offer dramatic visual proof that the dynamic, distributed advanced planning systems of the type envisioned in the Joint Vision 2010 and the Air Force's New World Vistas can be constructed today using commercial technology and our software architecture.